SrcTrace

Objective

Applications should always be traceable back to its source. When using a revision control system like git, this would mean being able to extract the source regardless of how old or home any revisions it has undergone since.

This projectlet generates a fragment of source code that can be compiled into the application and retrieved at runtime to report the details as necessary.

User needs and requirements

Id	Need/Requirement
1	The utility shall be able to report the specifications of the repository.
2	The utility shall be able to report the details - branch name, current commit id of the repository.
3	The utility shall be able to generate source fragments that can be compiled into an executable. The output can be in the programming languages: C, C++, Ada, go, python

User wants

The following will enhance the appeal of the tool.

Id	Wants
1	The utility shall be able to report on a specific file.

Specification

Command/Switch	Description
-major=<m>	Major version. Default=0
-minor=<n>	Minor version. Default=0
-build=<build>	Build number. Default=9999
-lang=()	List of languages separated by commas. Possible choices are: C, C++, Ada, python, Go,
-output=name	Filename of the output. Default is “revision” with a file type based on the language. E.g revision.h for C and revision.hpp for C++.

Command/Switch	Description
-cwd=<dir>	Set working directory to this before performing the function. Default is the current working dir
argument	A specific file name. By default, this will report on the current checked out repository.
Fields in the outputfile	VERSION_MAJOR VERSION_MINOR BUILD_NUMBER REPO_NAME BRANCH_NAME COMMIT_ID_FULL COMMIT_ID

Example usage